Gradiant Descent.

$f : R^n \longrightarrow R$

we need to $\min_{x \in R^n} f(x)$

we find the min iteratively by starting at init $x \Rightarrow x^0$

$$x_{k+1} = x_k + \alpha_k d_k$$

d_k : direction of improvment.

α_k : step size

d is direction of improvment if

$$\exists \alpha > 0 \quad s.t. \quad f(x + \alpha d) < f(x)$$

So gradient descent

$f : R^n \longrightarrow R$ differentiable then $(x_k)_{n \in N}$ sequence s.t $f(x_0) \geq f(x_1) \geq f(x_n) \cdots$

So this sequence is optainable by

$$x_{k+1} = x_k + \alpha d_k$$

where $d_k = -\nabla f(x_k) \quad \alpha_k > 0$

$$\alpha_k = \min_{\alpha_k \in R_n} f(x_k - \alpha_k \nabla f(x_k))$$

Subgradient Method

if $f: R^n \rightarrow R$ is not diffrentiable / continous

$$x_{k+1} = x_k + \alpha_k g_k$$

g_k is any subgradient of f at x_k

α_k pre determind value. before the run of the method.

S.M will Convergece slowly.

maybe α_k might increase the value of f

so $f_{k+1} = \min(f_k, f(x_k))$

Gradient descent for quadratic form:

$f: R^n \longrightarrow R \quad A \in R^n \times R^n$ symetric

$b \in R^n \quad c \in R$

$$f(x) = \frac{1}{2} x^T A x - x^T b + c$$

we know

$$\nabla f(x) = \frac{1}{2} Ax + \frac{1}{2} A^T x - b$$

$$\nabla f(x) = Ax - b$$

$H_f(x) = A \longrightarrow$ Positive defient

this means x^* $Ax^* - b = 0$ is a unique minimizer of f.

so $x^* = A^{-1} b$ we can't inverse A

we can compute α_k

$$x_{k+1} = x_k - \alpha_k \nabla f(x_k)$$

$$\alpha_k = \frac{d_k^T d_k}{d_k^T A d_k}$$

Example

$$A = \begin{pmatrix} 7.75 & 3.9 \\ 3.9 & 3.25 \end{pmatrix} \quad b = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad c = 0$$

$$f(x) = \frac{1}{2} x^T A x \quad \Rightarrow \nabla f(x) = Ax$$

$$H_f(x) = A$$

Starting $x_0 = \begin{pmatrix} 1 \\ -0.8 \end{pmatrix}$ $\quad \Rightarrow \nabla f(x_0) = A \begin{pmatrix} 1 \\ -0.8 \end{pmatrix} = \begin{pmatrix} 4.63 \\ 1.3 \end{pmatrix}$

$$x_1 = x_0 - \alpha_0 \nabla f(x_0)$$

$$x_1 = \begin{pmatrix} 1 \\ -0.8 \end{pmatrix} - 0.01 \begin{pmatrix} 4.63 \\ 1.3 \end{pmatrix} \Rightarrow \alpha_0 = 0.01$$

$$\begin{pmatrix} 1 \\ -0.8 \end{pmatrix} - \begin{pmatrix} 0.0463 \\ 0.013 \end{pmatrix} = \begin{pmatrix} 0.9537 \\ -0.813 \end{pmatrix} = x_1$$

$$\text{or } \alpha_0 = \frac{\nabla f(x_0)^T \nabla f(x_0)}{\nabla f(x_0)^T A \nabla f(x_0)} = \frac{23.1269}{228.716675}$$

$$\boxed{\alpha_0 = 0.10112}$$

$$\begin{pmatrix} 0.531 \\ -0.8013 \end{pmatrix} = x_1$$

better jump

diffrent α

the gradient is always ortnogonal to iso lines and show zigzagging characteristic.

Conjugateed Gradient.

A-orthogonal

$x, y \in \mathbb{R}^n$ are A-orthogonal or conjecated

energy norm $\langle x, y \rangle_A = 0$

$$\|x\|_A = \sqrt{\langle x, x \rangle_A} = \sqrt{x^T A x}$$

So in CG the idea is to choose the d_k A orthognal to be A-orthognal to each other.

the idea the same

$$x_{k+1} = x_k + \alpha_k d_k$$

$\forall i, j \mid i \neq j \quad \langle d_i, d_j \rangle_A = 0$ A-orthognal or Conjecateed.

$$\alpha_k = \min_{\alpha_k > 0} f(x_k + \alpha_k d_k)$$

residual $\boxed{r_k = b - A x_k}$ error step

error $\boxed{e_k = x^* - x_k}$ $\leftarrow$ the min

So $r_k = b - A x_k = A x^* - A x_k = A(x^* - x_k)$

$$\boxed{r_k = A e_k}$$

$$e_{k+1} = x_{k+1} - x^*$$

$$e_{k+1} = x_k + \alpha_k d_k - x^*$$

$$\boxed{e_{k+1} = e_k + \alpha_k d_k}$$

like Gradient descent

$$\boxed{\alpha_k = \frac{d_k^T r_k}{d_k^T A d_k}}$$

CG Can solve the minimization Problem in n steps giving the

$A \in R^{n \times n}$ Positive definent & symmetric.
and

$$f(x) = \frac{1}{2} x^T A x - x^T b + c$$

in this quadratic form.

this means $\boxed{e_n = 0}$ $\boxed{r_n = 0}$

#idea basically A change the space to (contourline to circles) so the gradient will go to max length

energy scalor product to change space. to circles.

Variants of CG:

CG is restricted to A s.p.d and f as quadratic function.

Problem if A is not s.p.d

- apply CG on normal equation

if $A \in R^{m \times n}$ $m > n$

we have $\min_{x \in R^n} \frac{1}{2} \|Ax - b\|_2^2$

normal equation $A^T A x = A^T b$

$\nabla f(x) = A^T (Ax - b)$ $H_f(x) = A^T A > 0$ Positive

if is not linear Function.

Solution : none linear CG / no convergence garanties.....

we have to accelerate convergent of CG1
we have huge matrix. so like 1m (n)
Solution : preconditioning

the first steps get the algo a lot further towards the solution.

we solve $M^{-1}Ax = M^{-1}b$

ideally $M = A \Rightarrow x = A^{-1}b$

$M = diag(A)$ so $M^{-1} = \begin{pmatrix} \frac{1}{a_{11}} & & \\ & \frac{1}{a_{22}} & \\ & & \frac{1}{a_{33}} \ddots \end{pmatrix}$

Tikhonov Regulaization:

let $Ax = b$ $\quad A \in R^{m \times n}$ $\quad b \in R^{m}$

$\xi \in R^{n}$ error

usually there is error

$$Ax + \xi = b \Rightarrow$$

ill condritionning: $m = n$ A is invertable

$$A^{-1}(Ax + \xi) = A^{-1}b$$
$$x + A^{-1}\xi = A^{-1}b$$
$$x = A^{-1}b - \underbrace{A^{-1}\xi}_{\text{evil}}$$

estimating the error size $\quad \|A^{-1}\xi\|_2 \leq \|A^{-1}\|_2 \underbrace{\|\xi\|_2}_{\text{this is small}}$

matrix norm

$$\|A\|_2 = \max_{x \neq 0} \frac{\|Ax\|_r}{\|x\|_2}$$

or $\quad \|A\|_2 = \sigma_{max}(A)$ max singular value.

$$\|Ax\| \leq \|A\| \; \|x\|$$

Condition of matrix: $\quad A \in R^{m \times n}$

$$cond(A) = \frac{\sigma_1(A)}{\sigma_n(A)} \leftarrow \text{singular value.}$$

$$\sigma_1 > \sigma_2 > \cdots > \sigma_n$$

$\sigma_1 = 1$ $\sigma_n \ll\ll 1$

So $\text{Cond}(A) = \sigma_{max}(A)$

so for $\text{cond}(A^{-1}) = \frac{1}{\sigma_n \text{ of } A} \gg\gg 1$

large error so the issue in A

So $\|A^{-1}\|$ is huge

in Tikonhof to reduce the the effect

to noice (explosion of errors).

So the biggest σ in $A = U \Sigma V^T$

$A^{-1} = (U \Sigma V^T)^{-1} = V \Sigma^{-1} U^T$

$$\Sigma = \begin{pmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_n \end{pmatrix}$$

$$\Sigma^{-1} = \begin{pmatrix} \frac{1}{\sigma_1} & & & \\ & \frac{1}{\sigma_2} & & 0 \\ & & \ddots & \\ 0 & & & \frac{1}{\sigma_n} \end{pmatrix}$$

this is not $U \Sigma V^T$ of A^{-1}

but they are the same in $A^{-1} = V \underline{\Sigma}^{-1} U^T$

not SVD

Cope with huge $cond(A^{-1})$

use trongated SVD remove all small singular values.

$$A = U \Sigma V^T \qquad \Sigma = diag(\sigma_1 \cdots \sigma_n)$$

approx.. $\hat{x} = \sigma_k(b) = V \Sigma_k^+ U^T b$

where $\Sigma_k^+ = diag\left(\frac{1}{\sigma_1}, \frac{1}{\sigma_2} \cdots \frac{1}{\sigma_k}, 0, 0, 0\right)$

We have $\sigma_k(b) = V \Sigma_k^+ U^T (Ax + \varepsilon)$

$$= V \Sigma_k^+ U^T U \Sigma V^T x + V \Sigma_k^+ U^T \varepsilon$$

$$= V \Sigma_k^+ \Sigma V^T x + \underbrace{V \Sigma_k^+ U^T \varepsilon}_{\text{error}}$$

$$\leq \| V \Sigma_k^+ U^T \| + \| \varepsilon \| \quad \leftarrow \text{small}$$

V and U^T are orthogonal are length preserving so we drop them

$$\| \Sigma_k^+ \|_2 = \sigma_{max} = \frac{1}{\sigma_k}$$

So the error is $= \frac{1}{\sigma_k} \| \varepsilon \|$

Since $\frac{1}{\sigma_1} < \frac{1}{\sigma_2} < \frac{1}{\sigma_k} \cdots$

more we remove the influenc of ε but also we reduce the solution correctness.

use Tikohof regulization to reduce the error influence.

Tikohnof Regulized solution.

$$T_\alpha(b) = \min \|Az - b\|^2 + \alpha \|z\|_2^2$$

for $\alpha > 0$ regulized parameter.

$$A = U \Sigma V^T$$

$$T_\alpha(b) = V D_\alpha^+ U^T b$$

$$D_\alpha^+ = diagg\left(\frac{\sigma_1}{\sigma_1^2+\alpha}, \dots \frac{\sigma_n}{\sigma_n^2+\alpha}\right)$$

make α bigger then make the quotient smaller so it doesn't explode the error.

Stack Form.

$$\begin{bmatrix} A \\ \sqrt{\alpha} I \end{bmatrix} x = \begin{bmatrix} b \\ 0 \end{bmatrix}$$

$$\tilde{A} x = \tilde{b}$$

$$\tilde{A}^T \tilde{A} x = \tilde{A}^T \tilde{b}$$

$$\begin{bmatrix} A^T & \sqrt{\alpha} I \end{bmatrix} \begin{bmatrix} A \\ \sqrt{\alpha} I \end{bmatrix} = A^T A + \alpha I$$

$$(A^TA + \alpha I)x = \tilde{A}^T\tilde{b}$$

$$\tilde{A}^T\tilde{b} = \begin{bmatrix} A^T & \sqrt{\alpha}I \end{bmatrix}\begin{bmatrix} b \\ 0 \end{bmatrix}$$

$$= A^Tb$$

so

$$\boxed{(A^TA + \alpha I)x = A^Tb}$$

$$T_\alpha(b) = (A^TA + \alpha I)^{-1}A^Tb$$

if A is full rank A^TA is symetric postive defient we can use CG to solve this.

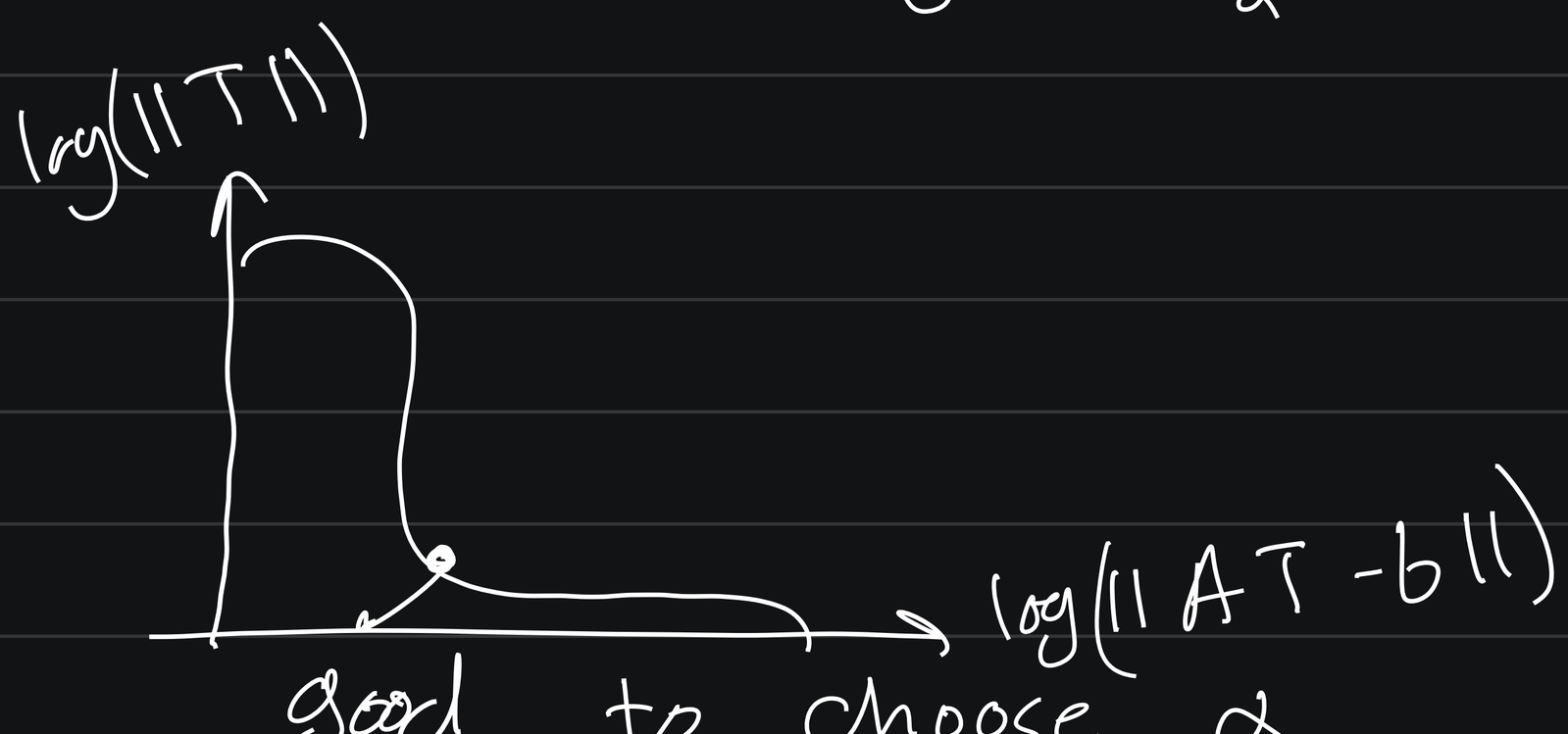
how to choose α

L curve method

we choose candidate of α

we compnt the solution for each α

2D curve $\left(\log \|A T_\alpha(6) - b\|, \log \|T_\alpha(b)\| \right)$



Newton Method

Solve system of non-linear Function

$\Omega \subset R^n, \quad F : \Omega \to R^n$

Solve

$$F(x) = 0$$

setting F to 0

start $x_0 \in \Omega \Rightarrow x_{k+1} = x_k - J_F^{-1}(x_k) F(x_k)$

Guess Newton Method (minimizing)

$g := R^n \to R^m$ solve $\min_{x \in R^n} \frac{1}{2} \| g(x) \|_2^2$

start $x_0 \in \Omega \Rightarrow x_{k+1} = x_k - \left(J_g(x_k)^T J_g(x_k) \right)^{-1} J_g(x_k)^T g(x_k)$.

Fixed point

let $N \subset M \neq 0$ $T := M \to N$

a solution x^* is called fixed point if

$T(x^*) = x^*$ doesn't change.

example

- $T(x) = x + 1$ $\longrightarrow$ no fixed points.

f.p.I does not converge

- $T_2(x) = ax$ $a \in R \neq 1$

if $a \neq 1$ only $x = 0$ $T(0) = a0 = 0$

so $x = 0$ fixed point.

f.p.I converges to 0 if $|a| < 1$ and
does n't not converge $|a| > 1$

- $T_3(x) = x$ every x is a fixed point.

Convergece at any x

Fixed point iteration

(M,d) $T: M \to M$ iteration

$x_0 \in M$ $x_{k+1} = T(x_k)$

will it lead to a fixed point

we will stop when $x_{k+1} = T(x_k) = x_k$

Lipschitz-continous / contraction

(M,d) $A \subset M$ $T: A \to M$ if $\exists L > 0$

s.t $d(T(x), T(y)) \leq L \cdot d(x,y)$
$\forall x, y \in A$

if $L \in (0,1)$ then we call it contraction.

Banach fixed point theorm.

let (X, d) $M \subset X$ $M \neq \emptyset$ $T: M \to M$

① complet space imp

② M closed

③ T is proper contraction

than $\exists x^*$ s.t (exactly one) s.t $x^* \in M$ and fixed point iteration convergec to it.

then $d(x_k, x^*) \leq \frac{L}{1-L} d(x_{k-1}, x_k)$

Brouwer

let $M \subset \mathbb{R}^n$

① Convex $T: M \to M$ continous

② Compact

③ $\neq \emptyset$

then there is at lecst one fixed point x^*

Fixed point iteration to solve non-linear equation

$F(x^*) = 0 \Leftrightarrow T(x^*) = x^*$

So $T(x) = x - J_F(x)^{-1} F(x)$.

important

fixed point iteration : everything depends on T

typical : $T(x) = x - F(x)$

so if $T(x^*) = x^*$

① then $T(x^*) = x^* - F(x^*)$

then $F(x^*) = 0 \implies$

② $T(x) = x - J_f^{-1}(x) F(x)$ newton method

F.P.I is the most generic type

of optimization.

General optimization problem

$$\min_x \; D(x, y) + \lambda R(x)$$

we can add multiple Regulization $+ \sum_{i=1}^{n} \lambda_i R_i(x)$

important question

if we see some optimization we need to choose which tool we can use for it to optimize it

for example if we have $\|x\|_1$ ← not differentiable

then GD, Does not work or newton doesn't also may be Subgradient method works.